

DETAILED ACTION

EXAMINER'S AMENDMENT

1. An Examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to the applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Douglas A. Dallmann, on 06/02/2009.

2. The application has been amended as follows: **In the claims:**

1. (Currently Amended) A method of generating identifier data for persistently identifying a user interface element of interest in a graphical user interface of a source computer program, the method comprising:

receiving data indicative of the user interface element of interest from a first software component; and

in response to receiving the data indicative of the user interface element of interest, generating an element path identifier of the user interface element of interest and uniquely identifying the user interface element of interest across different states of the first software component and returning at least the unique element path identifier to the first software component, the generating comprising storing exposed identifiers associated with selected ancestor nodes of a leaf node within a tree structure representation of the graphical user interface, the selected ancestor nodes located along a path of the tree structure representation from the leaf node to a designated root node, the selected ancestor nodes excluding at least one ancestor node of the leaf node located along the path that is uniquely identifiable from at least one first ancestor

node of the leaf node located along the path within a scope of a strongly named branch of the tree structure representation having a second ancestor node of the leaf node located along the path as the strongly named branch root node, the excluded at least one ancestor node being located between the strongly named branch root node and the at least one first ancestor node along the path, the exposed identifiers associated with the selected ancestor nodes being stored in the element path identifier, the leaf node representing the specified user interface element of interest;

wherein persistently and uniquely identifying the user element of interest comprises persistently and uniquely identifying the user interface element of interest across reboots of a computer running the source computer program.

2. (Original) The method of claim 1, wherein generating the element path identifier is implemented by a second software component communicative with the source computer program.

3. (Canceled)

4. (Currently Amended) The method of claim [[3]] 1, further comprising converting the element path identifier to a string type data structure.

5. (Canceled)

6. (Currently Amended) The method of claim [[3]] 1, wherein the exposed identifiers ~~information is a~~ are local alpha numeric identifiers not guaranteed to be unique, a class name, a module name associated with an application program or a sibling order.

7. (Canceled)

8. (Currently Amended) The method of claim [[7]] 1, further comprising:

storing in the element path identifier a strong name associated with the strongly named branch along with the named branch element identifier.

9-20. (Canceled)

21. (Currently Amended) At least one computer-readable medium having stored thereon computer-executable instructions related to a function responsive to a function call from a first software component, the function comprising:

an input parameter representing a user interface element of interest in a graphical user interface of a source computer program;

an output parameter representing an element path identifier for persistent unique identification of the user interface element of interest across multiple states of the source computer program, wherein the element path identifier comprises a hierarchical path of inheritance from the user interface element of interest to a parent root element; and

executable software for receiving the input parameter representing a user interface element of interest and in response, generating the output parameter representing an element path identifier of the user interface element of interest such that the output parameter represents an identifier capable of persistently identifying the user interface element of interest across different builds of the source computer program, the generating comprising storing exposed identifiers associated with selected ancestor nodes of a leaf node within a tree structure representation of the graphical user interface, the selected ancestor nodes located along a path of the tree structure representation from the leaf node to a parent root node representing the parent root element, the selected ancestor nodes not including at least one ancestor node of the leaf node located along the path that is uniquely identifiable from at least one first ancestor node of the leaf node located along the path within a scope of a strongly named branch of the tree structure representation having a second ancestor node of the leaf node located along the path as the strongly named branch root node, the excluded at least one ancestor node being located between the strongly named branch root node and the at least one first ancestor node along the path, the exposed identifiers associated with the selected ancestor nodes being stored in the element path identifier

represented by the output parameter, the leaf node representing the user interface element of interest.

22. (Original) The computer-readable medium of claim 21, wherein the element path identifier is a string type data structure.

23. (Canceled)

24. (Currently Amended) The computer-readable medium of claim 21, wherein the element path identifier comprises class names of component elements of an element path in the tree structure representation and related to the element of interest.

25. (Currently Amended) The computer-readable medium of claim 21, wherein the element path identifier comprises a module name of an application program related to component elements of an element path in the tree structure representation and related to the element of interest.

26. (Original) The computer-readable medium of claim 21, wherein the element path identifier comprises sibling order data.

27. (Canceled).

28. (Currently Amended) At least one computer-readable medium having stored thereon computer-executable instructions related to a function responsive to a function call from a first software component, the function comprising:

an input parameter representing an element path identifier of a target user interface element in a graphical user interface of a target computer program, wherein the element path identifier comprises a functional hierarchical path of inheritance from the target user interface to a parent root element;

an output parameter representing a location of the target user interface element within a hierarchical tree structure representation of the graphical user interface; and

executable software for receiving the element path identifier of the target user interface element and determining the output parameter representing the location of the target user interface element such that the output parameter represents an identifier capable of persistently and uniquely identifying the target user interface element across different builds of the target computer program, the determining comprising storing exposed identifiers associated with selected ancestor nodes of a leaf node within the hierarchical tree structure representation of the graphical user interface, the selected ancestor nodes located along a path of the hierarchical tree structure representation from the leaf node to a parent root node representing the parent root element, the selected ancestor nodes not including at least one ancestor node of the leaf node located along the path that is uniquely identifiable from at least one first ancestor node of the leaf node located along the path within a scope of a strongly named branch of the hierarchical tree structure representation having a second ancestor node of the leaf node located along the path as the strongly named branch root node, the excluded at least one ancestor node being located between the strongly named branch root node and the at least one first ancestor node along the path, the exposed identifiers associated with the selected ancestor nodes being stored in the identifier represented by the output parameter, the leaf node representing the target user interface element.

29. (Original) The computer-readable medium of claim 28, wherein the element path identifier is a string type data structure.

30. (Original) The computer-readable medium of claim 28, wherein the element path identifier comprises exposed identifier information of component elements of an element path related to the element of interest.

31. (Previously Presented) The computer-readable medium of claim 28, wherein the element path identifier comprises class names of component elements of an element path related to the element of interest.

32. (Previously Presented) The computer-readable medium of claim 28, wherein the element path identifier comprises a module name of an application program related to component elements of an element path related to the element of interest.

33. (Original) The computer-readable medium of claim 28, wherein the element path identifier comprises sibling order data.

34. (Canceled)

35. (Currently Amended) In a computer system running a computer program with a graphical user interface, a system for generating persistent, unique element path identifiers of elements of the graphical user interface and later searching for the elements of the graphical user interface using the element path identifiers, the system comprising:

a memory storing computer-executable instructions, the instructions comprising:

an API module comprising a first set of APIs related to passing function calls for generating the element path identifiers and a second set of APIs related to passing function calls for searching for the elements of the graphical user interface using the element path identifiers; and

an element path engine responsive to the function calls for generating the element path identifiers and to the function calls for searching for the elements of the graphical user interface using the element path identifiers, wherein the element path identifiers comprise information relating to functional ancestor elements and sibling elements of the elements of the graphical user interface such that the element path identifiers persistently and uniquely identify the user interface elements across a build of the computer program, at least one of the element path identifiers comprising exposed identifiers associated with selected ancestor nodes of a leaf node within the a tree structure representation of the graphical user interface, the selected

ancestor nodes located along a path of the tree structure representation from the leaf node to a parent root node, the selected ancestor nodes excluding at least one ancestor node of the leaf node located along the path that is uniquely identifiable from at least one first ancestor node of the leaf node located along the path within a scope of a strongly named branch of the hierarchical tree structure representation having a second ancestor node of the leaf node located along the path as the strongly named branch root node, the excluded at least one ancestor node being located between the strongly named branch root node and the at least one first ancestor node along the path, the leaf node representing an element of the graphical user interface; and a processor for executing the stored instructions.

36. (Previously Presented) The method of claim 1, wherein persistently and uniquely identifying the user element of interest further comprises persistently and uniquely identifying the user interface element of interest across different builds of the source computer program.

37. (Previously Presented) The computer-readable medium of claim 21, wherein the output parameter further represents an identifier capable of persistently identifying the target user interface element across different builds of the source computer program.

38. (Previously Presented) The computer-readable medium of claim 21, wherein the element path identifier identifies the source computer program in which the user interface element resides.

39. (Previously Presented) The computer-readable medium of claim 21, wherein the element path identifier identifies a user interface platform used to create the user interface element.

40. (Previously Presented) The computer-readable medium of claim 21, wherein the element path identifier identifies an application frame in which the user interface element resides.

41. (Previously Presented) A method, comprising:

- receiving data indicative of a specified user interface element in a graphical user interface of a computer program; and
- generating an element path identifier of the specified user interface element, the generating comprising:
 - locating a leaf node within a tree structure representation of the graphical user interface, the located leaf node representing the specified user interface element;
 - storing an identifier associated with the located leaf node in an element path identifier data structure; and
 - storing identifiers associated with selected ancestor nodes of the located leaf node located along a path of the tree structure representation from the located leaf node to a designated root node, the selected ancestor nodes excluding at least one ancestor node of the located leaf node located along the path that is uniquely identifiable from at least one first ancestor node of the located leaf node located along the path within a scope of a strongly named branch of the tree structure representation having a second ancestor node of the located leaf node located along the path as the strongly named branch root node, the excluded at least one ancestor node being located between the strongly named branch root node and the at least one first ancestor node along the path, the identifiers associated with the selected ancestor nodes being stored in the element path identifier data structure;
- wherein the element path identifier is a composite identifier comprised of the stored identifiers associated with the located leaf node and the stored identifiers associated with the selected ancestor nodes and the element path identifier persistently and uniquely identifies the specified user interface element across reboots of the computer program, across different states of the computer program and across different builds of the computer program.

Allowable Subject Matter

3. Claims 1-2, 4, 6, 8, 21-22, 24-26, 28-33, 35-41 are allowed.
4. The following is an examiner's statement of reasons for allowance:
5. The current application is directed to a method, a system and a computer readable memory for providing an identifier information generator for a user interface element of interest within a user interface of a computer program based on a description of a hierarchical element comprising parent elements that the user interface element inherits from, wherein the user interface element of interest is uniquely and persistently identifiable across different states of the computer program.

The closest art combination to the current application is De Armas et al. (De Armas, US 5,864,819) and Mir Farooq ali et al. ("Mir", Building multiplatform user interfaces with UIML). De Armas in view of Mir discloses receiving data indicative of a user interface element of interest within a computer program, generating an element path identifier of the user interface element of interest within a computer program for persistently and uniquely identifying the user interface element of interest across different states of the computer program. De Armas and Mir do not disclose "storing identifiers associated with selected ancestor nodes of the located leaf node located along a path of the tree structure representation from the located leaf node to a designated root node, the selected ancestor nodes excluding at least one ancestor node of the located leaf node located along the path that is uniquely identifiable from at least one first ancestor node

of the located leaf node located along the path within a scope of a strongly named branch of the tree structure representation having a second ancestor node of the located leaf node located along the path as the strongly named branch root node, the excluded at least one ancestor node being located between the strongly named branch root node and the at least one first ancestor node along the path, the identifiers associated with the selected ancestor nodes being stored in the element path identifier data structure". These limitations, in specific combination as recited in independent claims, define patentability of the claims over prior art of record.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to HAOSHIAN SHIH whose telephone number is (571)270-1257. The examiner can normally be reached on m-f 0730-1700.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kieu Vu can be reached on (571) 272-4057. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Kieu Vu/

Supervisory Patent Examiner, Art Unit 2173